illustrated in FIGS. **6**a, **6**b, and **6**c. Although all three options satisfy the lossless requirement described above, error-recovery is difficult when brute-force fragmentation is used. Therefore, a minimal number of bits are used to satisfy the lossy requirement described above.

TABLE 1

Options for Fragment-header syntax for brute-force fragmentation

Fragment-header syntax	Description	Overhead	Advantages	Disadvantages
Option 0: Start flag, End flag	Start flag is set in the first fragment of a sample. End flag is set in the last fragment of the same sample.	2 bits	Low overhead Easy to parse	Does not help in error recovery
Option 1: Sample ID	All fragments of one sample share the same Sample ID.	4 bits	Easy to parse	Does not help in error recovery
Option 2: TotalFragments PerSample	Each fragment contains the total number of fragments in the sample.	4 bits	Helps the receiver in error recovery.	

[0053] FIG. **6**a shows an RTP fragment packet **628**, where the fragment header **652** comprises a 2-bit binary syntax identifier **660**, indicating option 0, start and end flag fields **662**, and reserved field **668**. FIG. **6**b shows the RTP fragment packet **628**, where the fragment header **652** in this case comprises a 2-bit binary syntax identifier **660**, indicating syntax option 1, a sample ID field **664**, and a reserved field **668**. FIG. **6**c shows the RTP packet **628**, where the fragment header **652** in this case comprises a 2-bit binary syntax identifier, indicating syntax option 2, a TotalFragmentsPerSample field **666**, and a reserved field **668**.

[0054] Focusing on the third syntax option, for error recovery, a receiver can first identify the missing fragments from the syntax of the received fragments. As shown in Table 1, among the three options, the third syntax helps the receiver in determining the fraction of the missing fragments. The receiver may then decide whether to request retransmission of any missing fragment packets or to perform error-concealment by engaging in post-processing. Although sequence numbers associated with each RTP packet allows the receiver to know the proper ordering of the RTP packets, and consequently, whether any RTP packet is missing, they do not inform the receiver of which particular XML sample any one fragment is a part. Therefore, in addition to sequence numbers, the total number of fragments that comprise a sample is also provided with each fragment. If packet loss occurs, the receiver can correctly estimate how many fragments of a particular sample are in fact missing. In addition, the P flag in the common payload header informs the receiver what the priorities of the missing fragments are, while the TotalFragmentsPerSample informs how much percentage of a given sample is lost. Hence, these two types of information at different granularities, help facilitate selective retransmission of any lost fragment packets.

[0055] FIGS. **3** and **4** illustrate how information associated with fragments can be used to determine packet loss and what the receiver could decide to do in such a packet loss event. In particular, FIG. **3** shows one method of identifying a packet in the event of packet loss in brute force XML fragmentation. A sample 1 is shown, the content of which is partitioned into RTP transport packets **300-340**, each containing a fragment of sample 1. The RTP transport packets **300-340** are each associated with an RTP sequence number, i.e., 1-5, respectively. As described above, when the third syntax of brute force XML fragmentation is utilized, the fragment header of each of the RTP transport packets **300-340** contain a TotalFragmentsPerSample field indicating the total number of fragments into which a sample was partitioned. Here, the total number of fragments is five. In addition, each of the RTP transport packets **300-340** have an equal priority of 2. If, for example, RTP transport packet **310** was lost, the receiver can determine this packet loss from the RTP sequence numbers. Therefore, because the receiver knows that the second RTP transport packet **310** is missing, and it knows that it is one of five fragments of sample 1, with priority of 2, the missing RTP transport packet **310** can either be retransmitted. Alternatively, error correction may be performed at the receiver since the majority of the sample 1 packets have arrived. It should be noted that the RTP sequence number and the P flag are already defined in the generic SVG RTP packet format of FIG. **5**a, described above.

[0056] FIG. **4** shows another method of identifying a group of packets in the event of packet loss in brute force XML fragmentation. In this scenario, two samples, sample 1 and sample 2 are shown, where the content of sample 1 is partitioned into three fragments, each fragment being contained in an RTP transport packet, **400-420** respectively. Sample 2, on the other hand, is partitioned into 4 fragments, each of which is contained in at RTP transport packet **430-460** respectively. RTP sequence numbers 1-7 are assigned to each of the RTP transport packets **400-460**, where the RTP transport packets **400-420** each have a TotalFragmentsPerSample value of 3, while each of the RTP transport packets **430-460** have a TotalFragmentsPerSample value of 4. Lastly, the RTP transport packets **400-420** are each associated with a sample priority of 3, while the RTP transport packets **430-460** are each associated with a sample priority of 2.

[0057] As described with regard to FIG. **3**, from the RTP sequence numbers a receiver can determine that the second, third, and fourth RTP transport packets, i.e., **410**, **420**, and **430**, are missing. From the total fragments in each sample, the receiver can also determine that the first two missing RTP transport packets **410** and **420** belong to sample 1. In addition, the receiver knows the RTP transport packets **410**